

Creating:-

app.component.html:

```
<div>
  <form (ngSubmit)="onSubmit(dat)" #dat="ngForm">
    <div class="form-group">
      <label>Name </label>
      <input type="text" class="form-control"
name="name" [(ngModel)]="name1">
    </div>
    <div class="form-group">
      <label>Email </label>
      <input type="text" class="form-control"
name="email" [(ngModel)]="email1">
    </div>
```

</table> Phone </table>

<input type="text" class="form-control" [(ngModel)]=

</div> "phone1">

<button class="btn btn-primary"> Create </button>

</form>

</div>

variables declared in  
typescript file.

app.component.ts

onSubmit (data)

{

var newData = data.value

console.log(newData)

}

o/p:-

name: nivedita

email: nivedita@gmail.com

phone: 7894561230

post(): post method is used to insert the data  
in json server it expects two argument.  
first argument url second argument values.

onSubmit (data) { console.log (data); console.log (data.value)

this.http.post (this.url, data.value).subscribe ()

}

o/p:- without refreshing.

Delete	blesina: blesina@gmail.com
--------	----------------------------

Delete	akshata: akshata@gmail.com
--------	----------------------------

Name

niveditha

Email

niveditha@gmail.com

mobile

9876543210

o/p2: After refreshing.

Delete	blesina: blesina@gmail.com
--------	----------------------------

Delete	akshata: akshata@gmail.com
--------	----------------------------

Delete	niveditha: niveditha@gmail.com
--------	--------------------------------

\* By refreshing the angular project the entire json data will be reflected on the web-page.

\* By default json server will add the id. All the id's has to be unique

~~onSubmit~~

app.component.ts

onSubmit(data)

```
{  
  let newData = data.value  
  this.http.post(this.url, data.value).subscribe((res:  
    Details) => { this.data.push(newData) }  
  })  
}
```

O/p:-

Delete	blessina : blessina@gmail.com
Delete	akshatha : akshatha@gmail.com
Delete	nuvadiha : nuvadiha@gmail.com

Name

nuvadiha

Email

nuvadiha@gmail.com

Mobile

7987654321

onSubmit(data)

```
{  
  let newData = data.value  
  this.http.post(this.url, data.value).subscribe((res) => {  
    this.data.unshift(newData)  
  })  
}
```

onSubmit (data)

{

this.http.post(this.url, data.value).subscribe(res =>

{ this.data.push(newData)

data.reset()

})

}